

FIG. 1

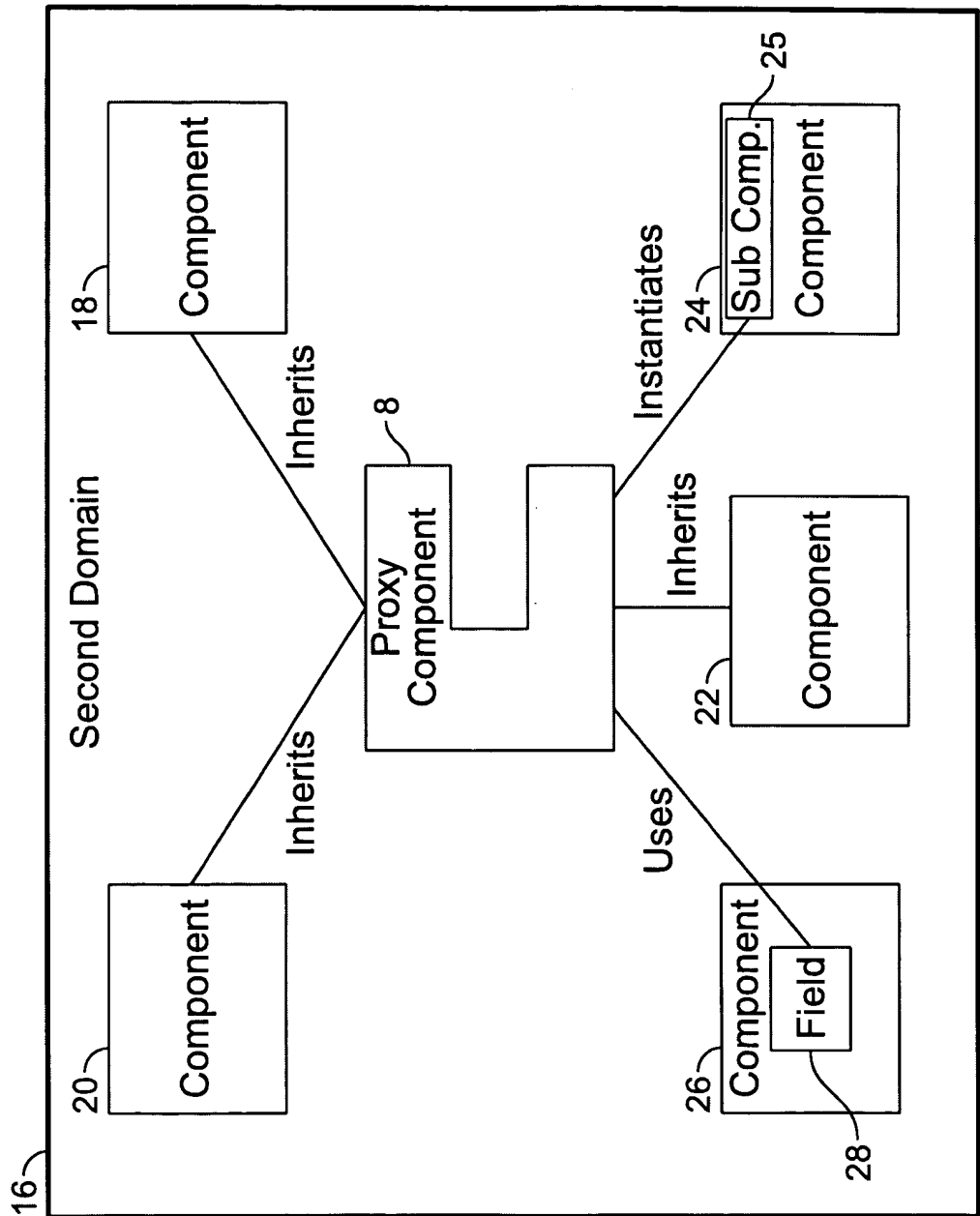
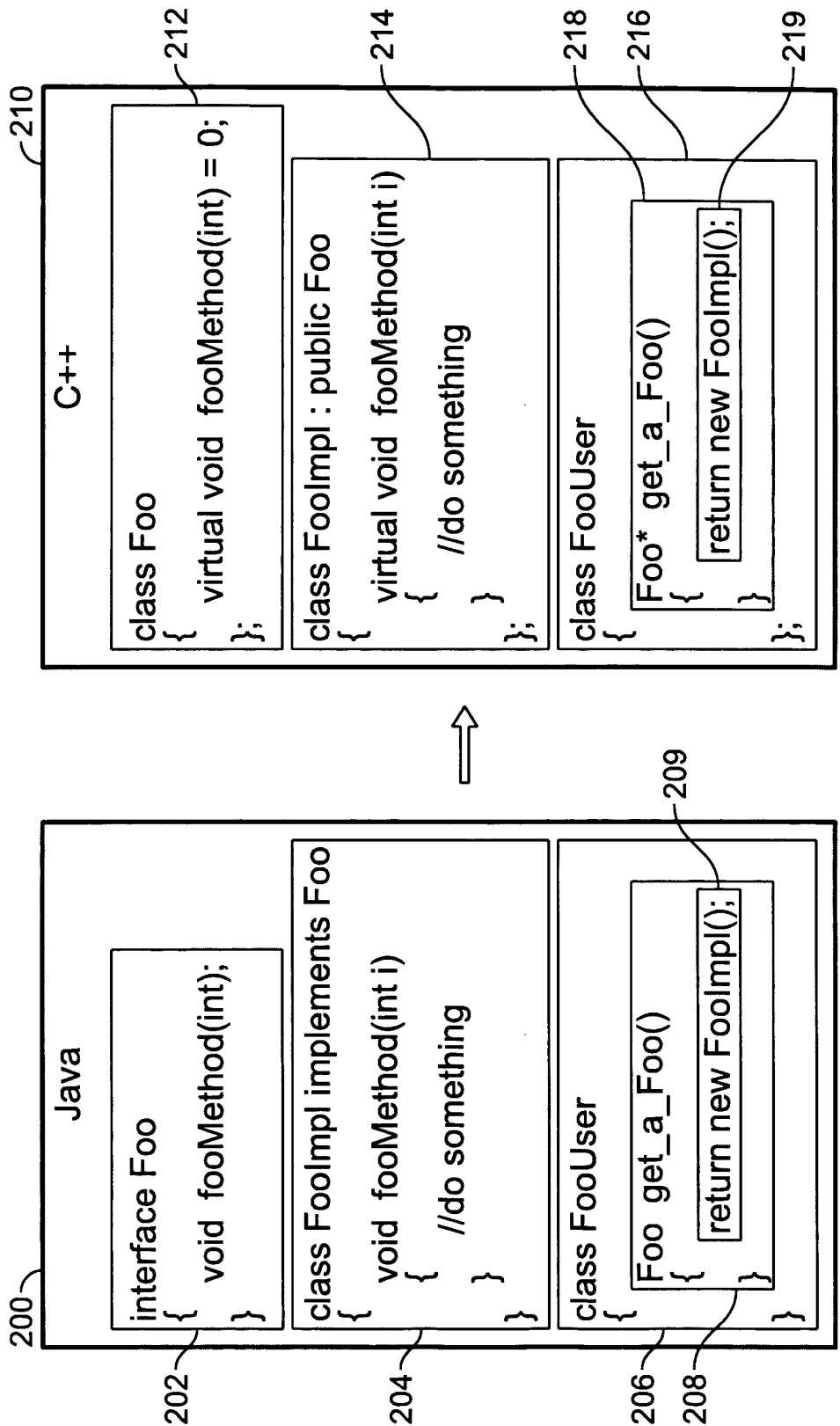


FIG. 2



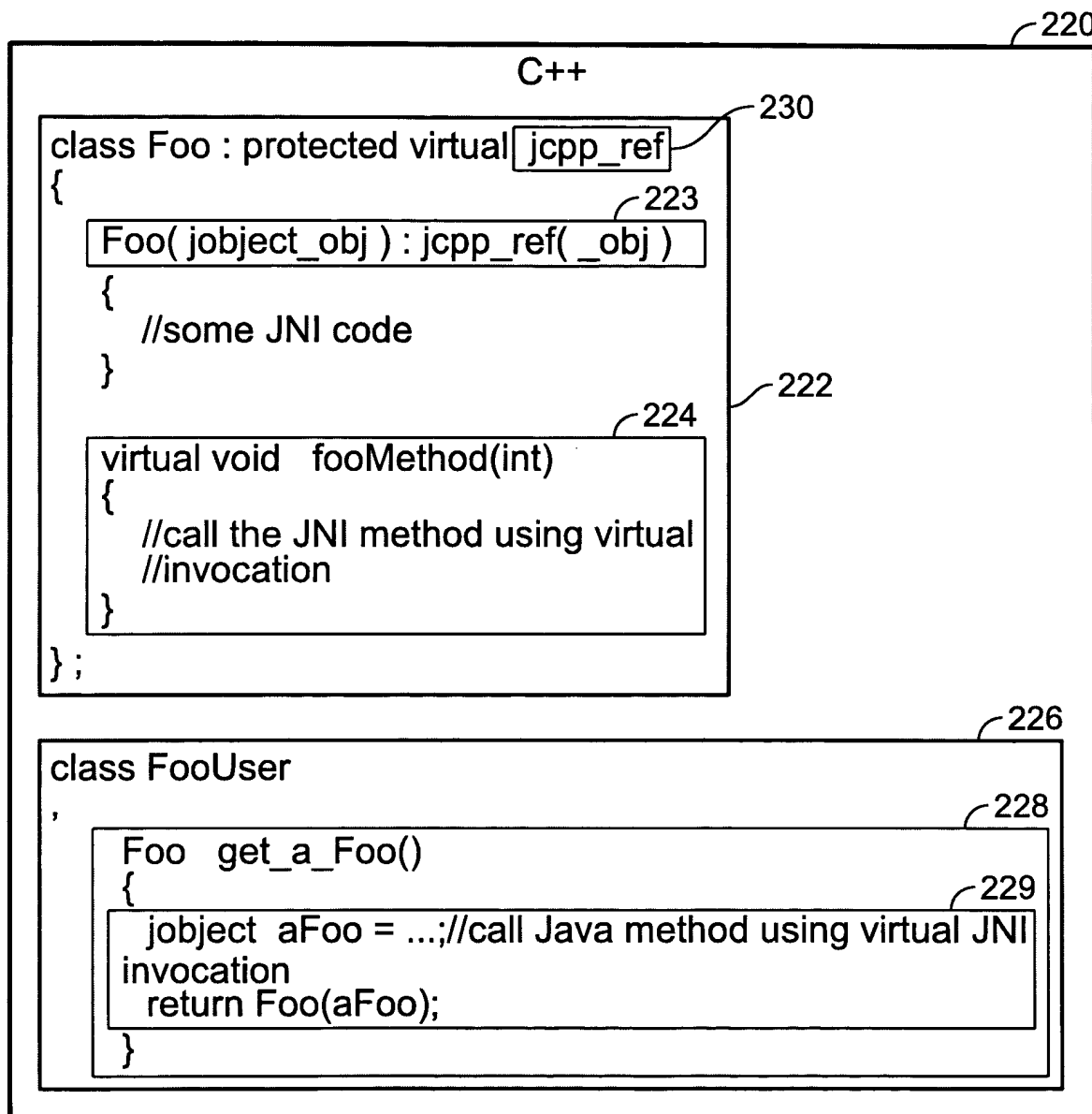


FIG. 5



30  
proxy\_name (jobject\_obj, int\_type);

36 38 40  
proxy\_name (const proxy\_ref\* \_ref, const char\* \_fieldName);

42 44 46  
proxy\_name (const proxy\_class\* \_clazz, const char\* \_fieldName);

48 50 52  
proxy\_name (const proxy\_array\* \_array, jsize \_index);

FIG. 6



108

```

class Foo : public java::lang::Object
{
public:
    typedef jcpp_object_array<Foo> array1D;
    Foo( const Tnull B);
    Foo( object, int );
    Foo( const jcpp_ref*, const char*);
    Foo( const jcpp_class*, const char*);
    Foo( const jcpp_array*, jsize);
    Foo( const Foo &);
    ~Foo();
    Foo& operator = ( const Foo &);
    bool operator == ( const Foo &) const;
    bool operator != ( const Foo &) const;
    static const jcpp_class* get_static_class();
    const jcpp_class* get_class() const;
    static Foo dyna_cast( const jcpp_ref & _src );
};
  
```

110  
112  
114  
116  
118  
120  
122  
124  
126  
128  
130

FIG. 7

52

```

class jcpp_int : public jcpp_base
{
public:
    typedef jcpp_int_array array1D;
    typedef (Tobject_array<array1D>) array2D;

    jcpp_int( const jcpp_ref * _ref, const char * _fieldName );
    jcpp_int( const jcpp_class * _ref, const char * _fieldName );
    jcpp_int( const jcpp_int_array * _array, jsize _index );
    jcpp_int( const jcpp_int & _rns );
    jcpp_int( );
    operator new ( size_t _size );
    operator delete( void * _ptr );
    operator jint ( ) const;

    jcpp_int & operator = ( jint );
    jcpp_int & operator += ( jint );
    jcpp_int & operator -= ( jint );
    jcpp_int & operator *= ( jint );
    jcpp_int & operator /= ( jint );
    jcpp_int & operator %= ( jint );
    jcpp_int & operator ++ ( );
    jcpp_int & operator -- ( );
    jint operator ++ ( int );
    jint operator -- ( int );

    const jcpp_class * get_class() const;
};
    
```

54 56 57 58 60 61 62 59 65 66 62 64 66 68

FIG. 8

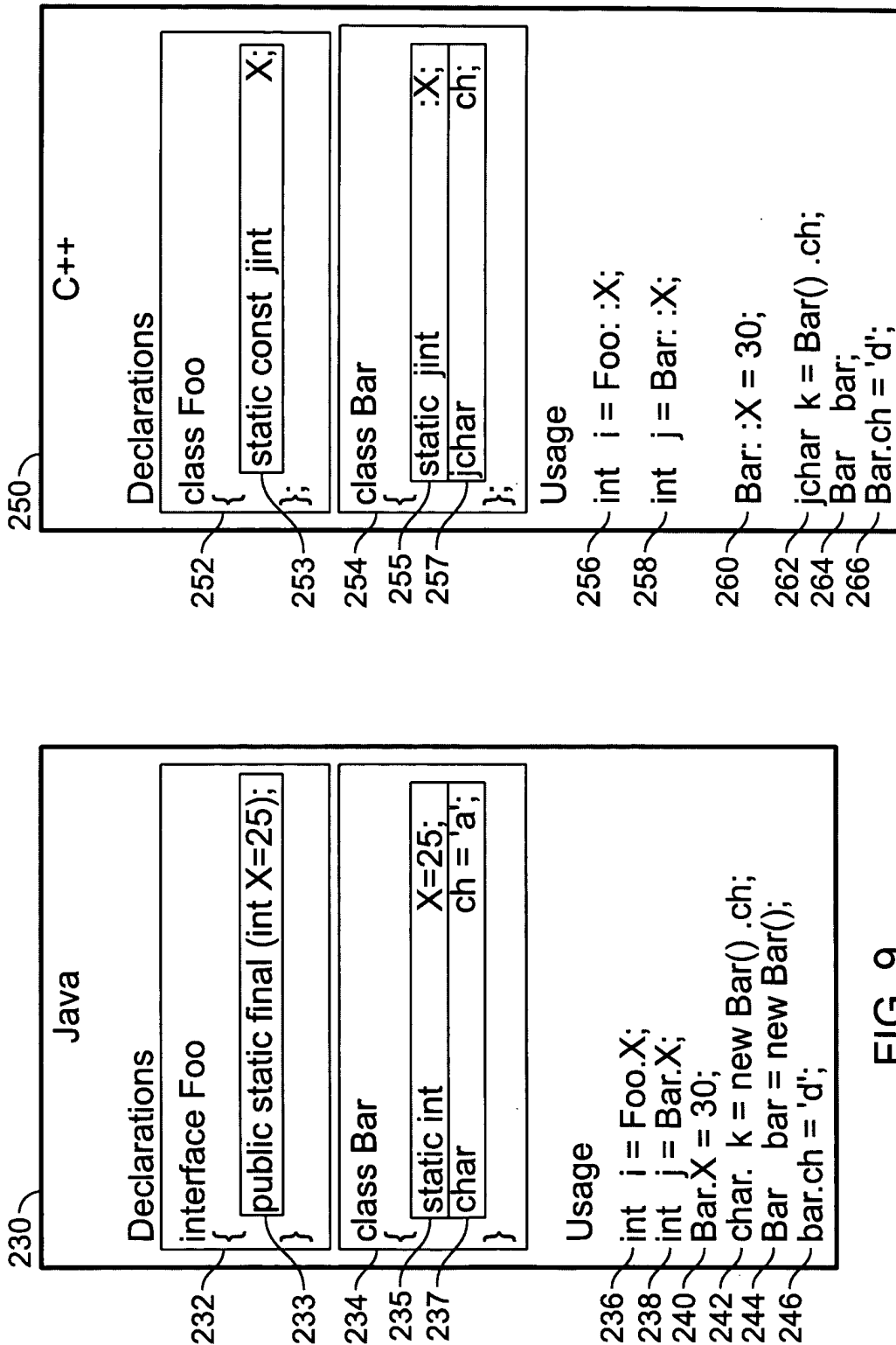


FIG. 9

FIG. 10



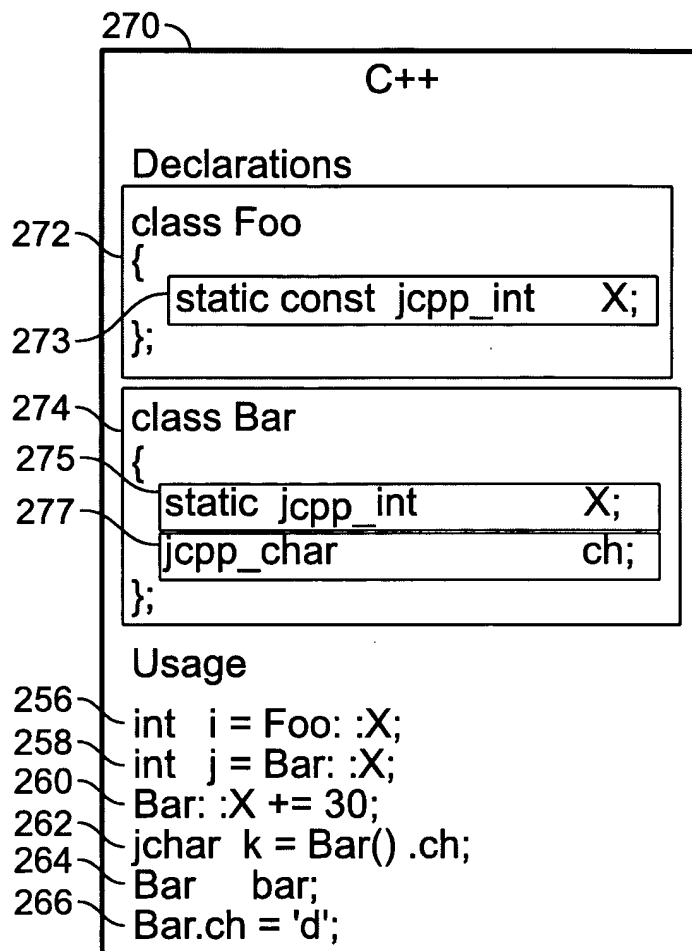
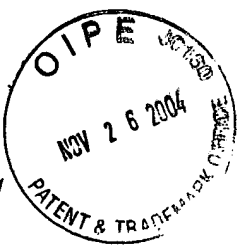


FIG. 11

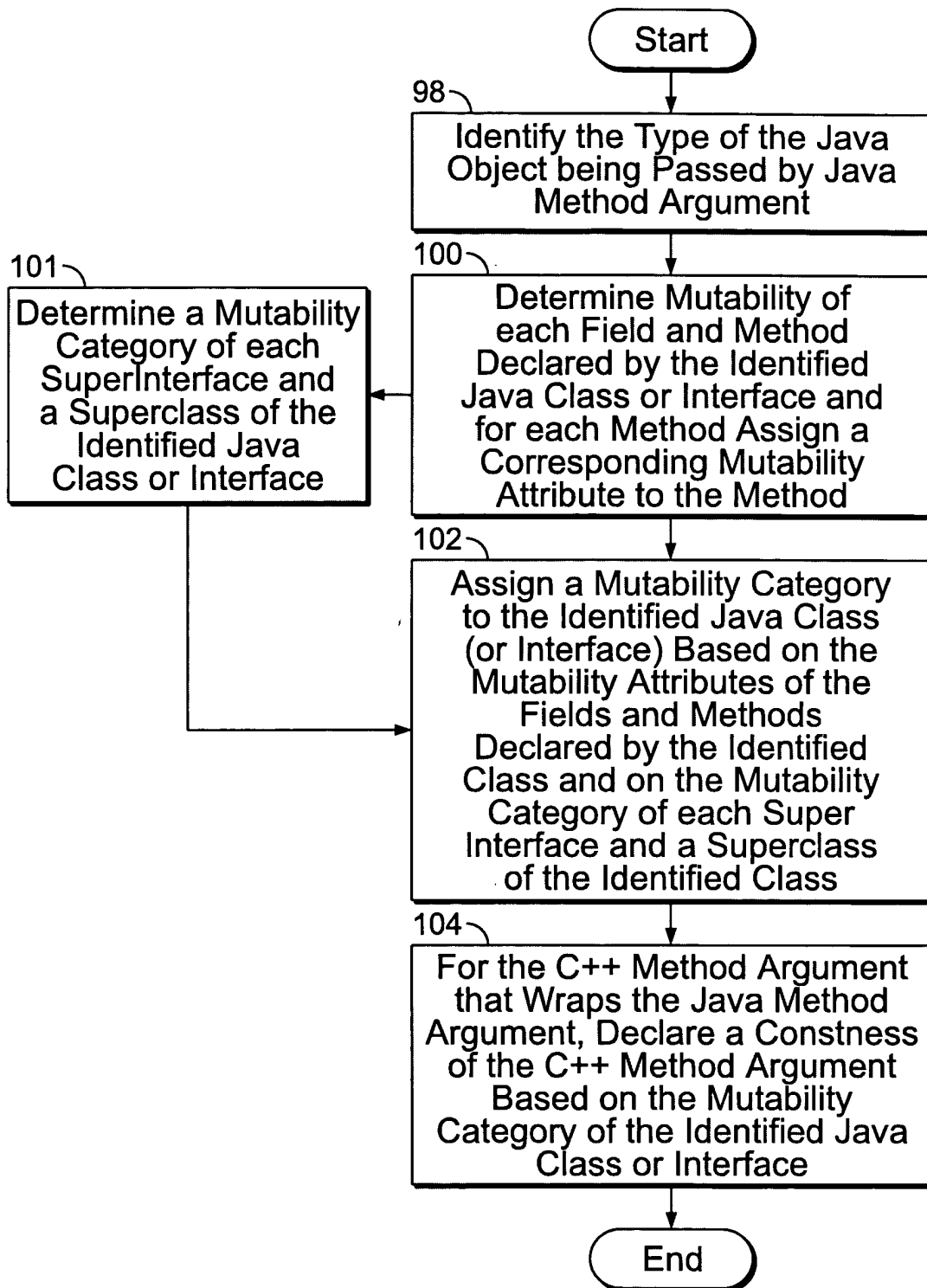


FIG. 12

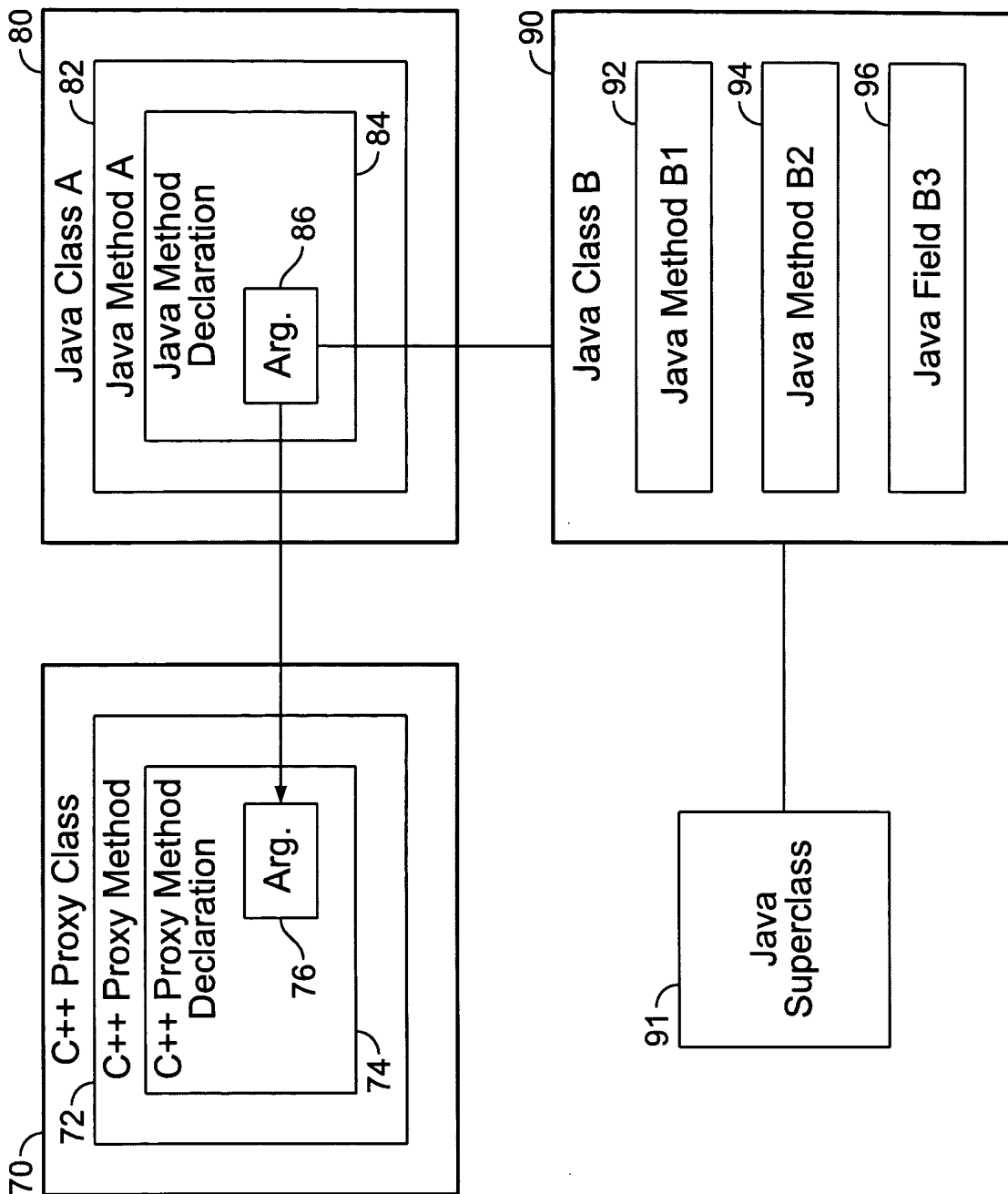


FIG. 13

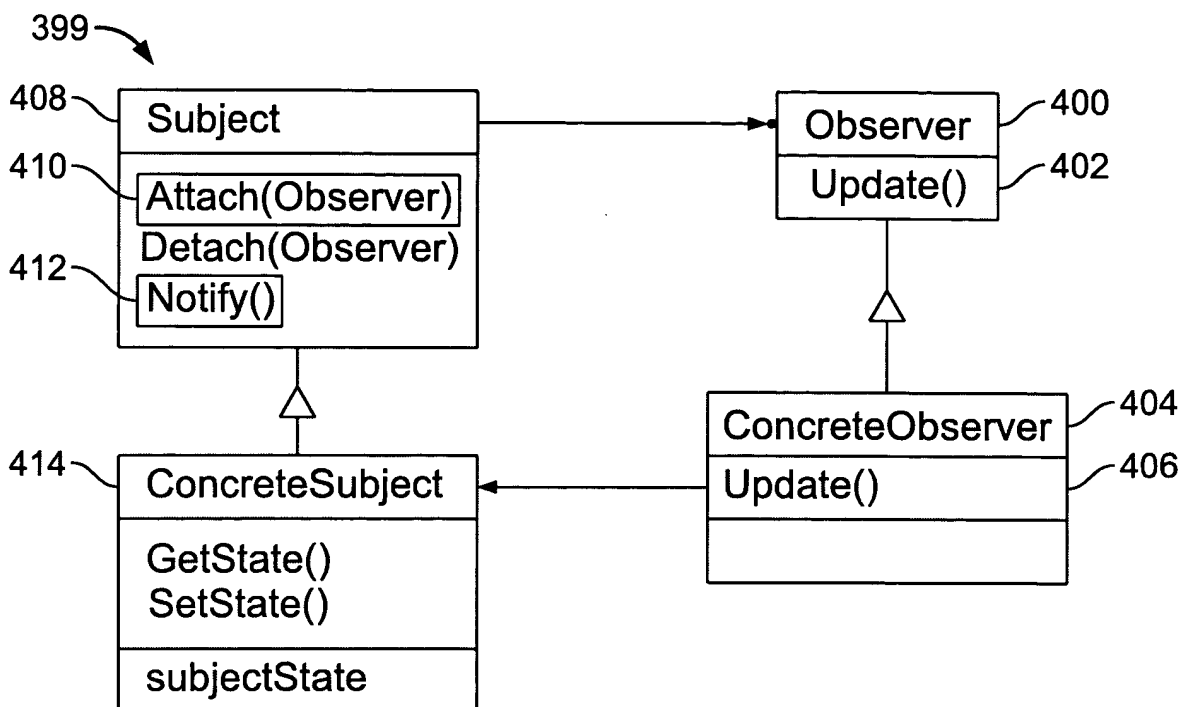


FIG. 14

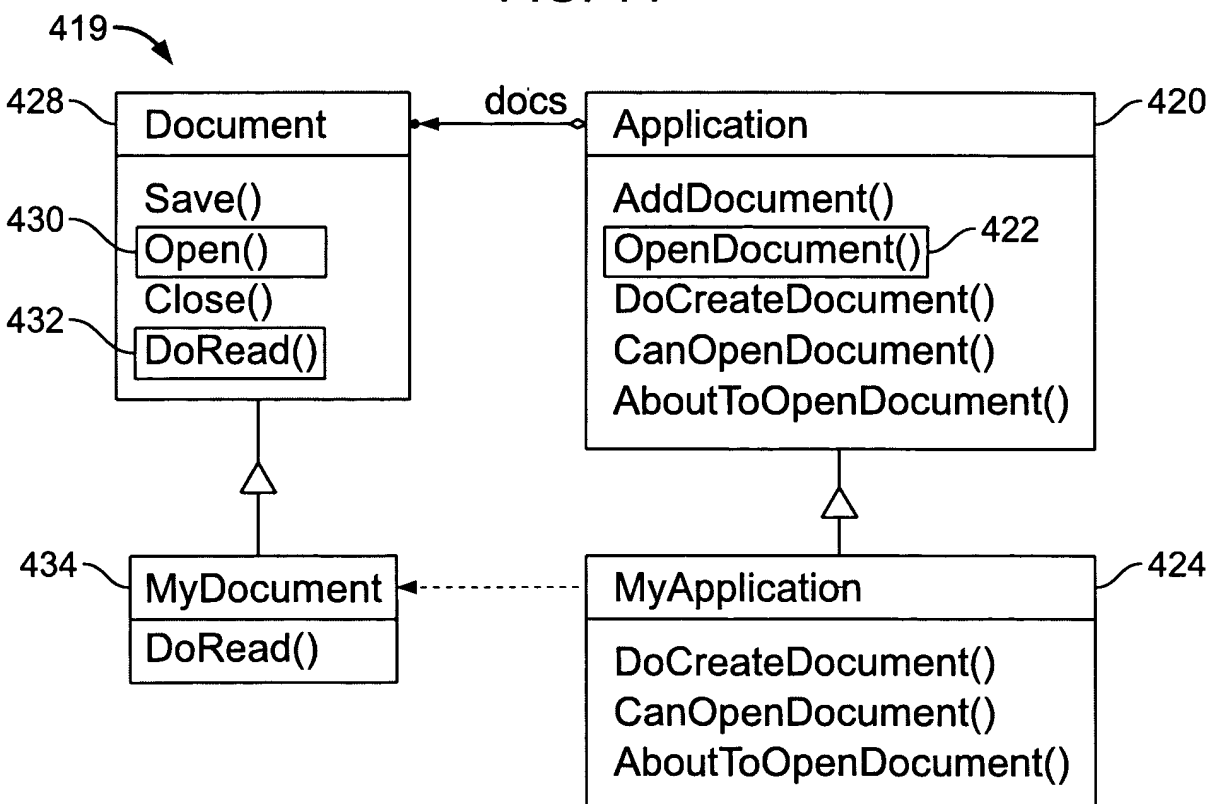


FIG. 15

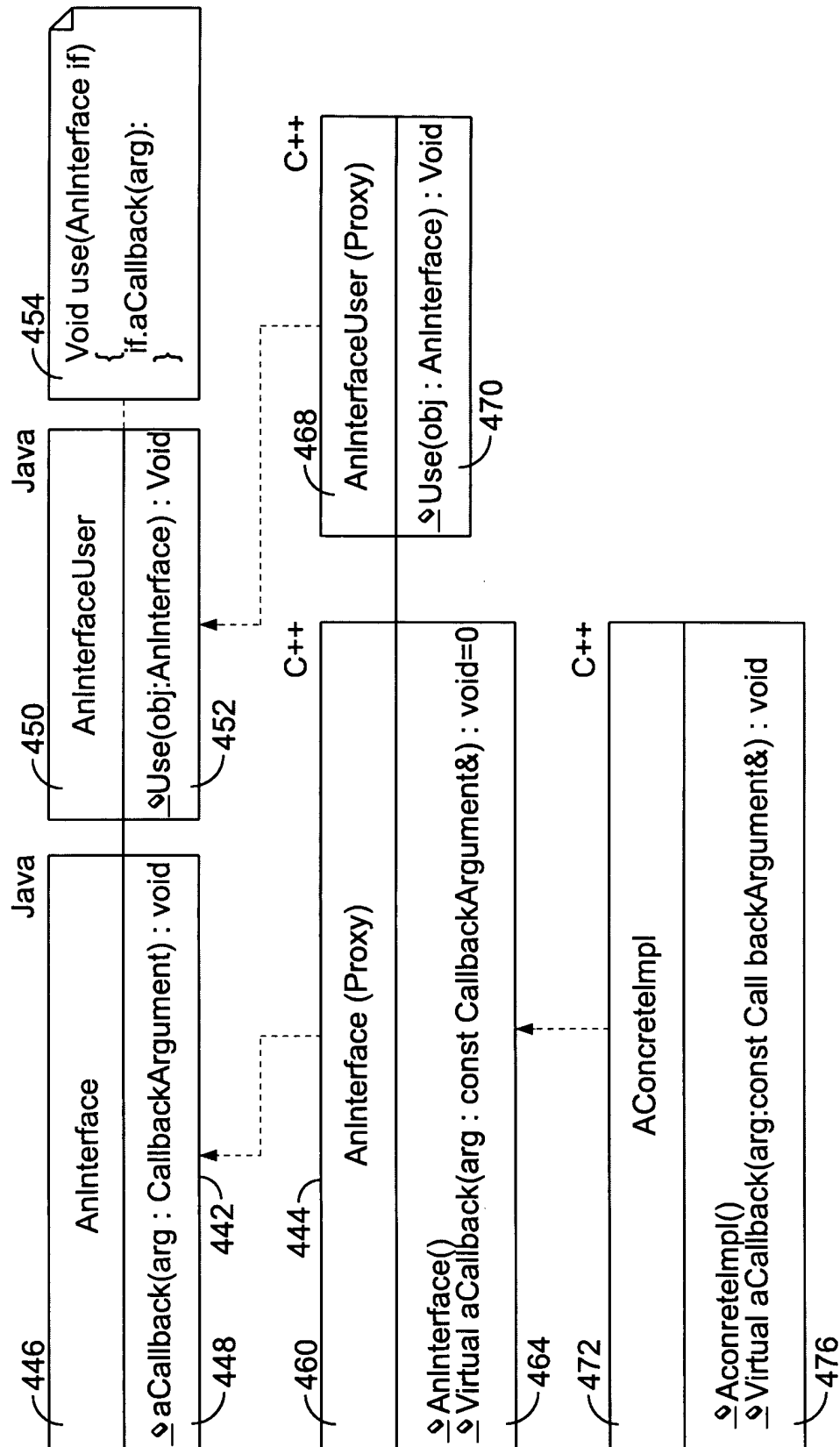


FIG. 16

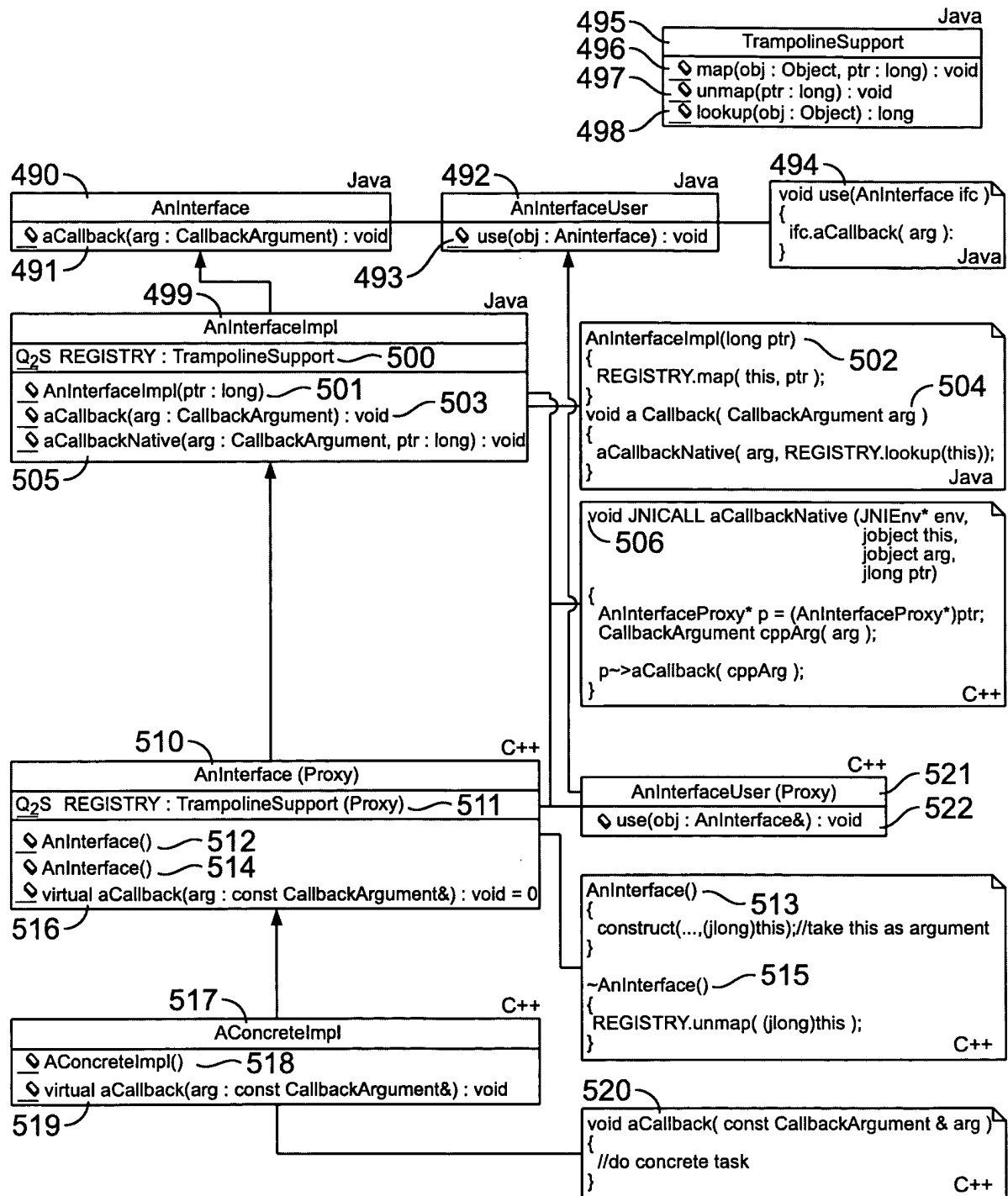


FIG. 17

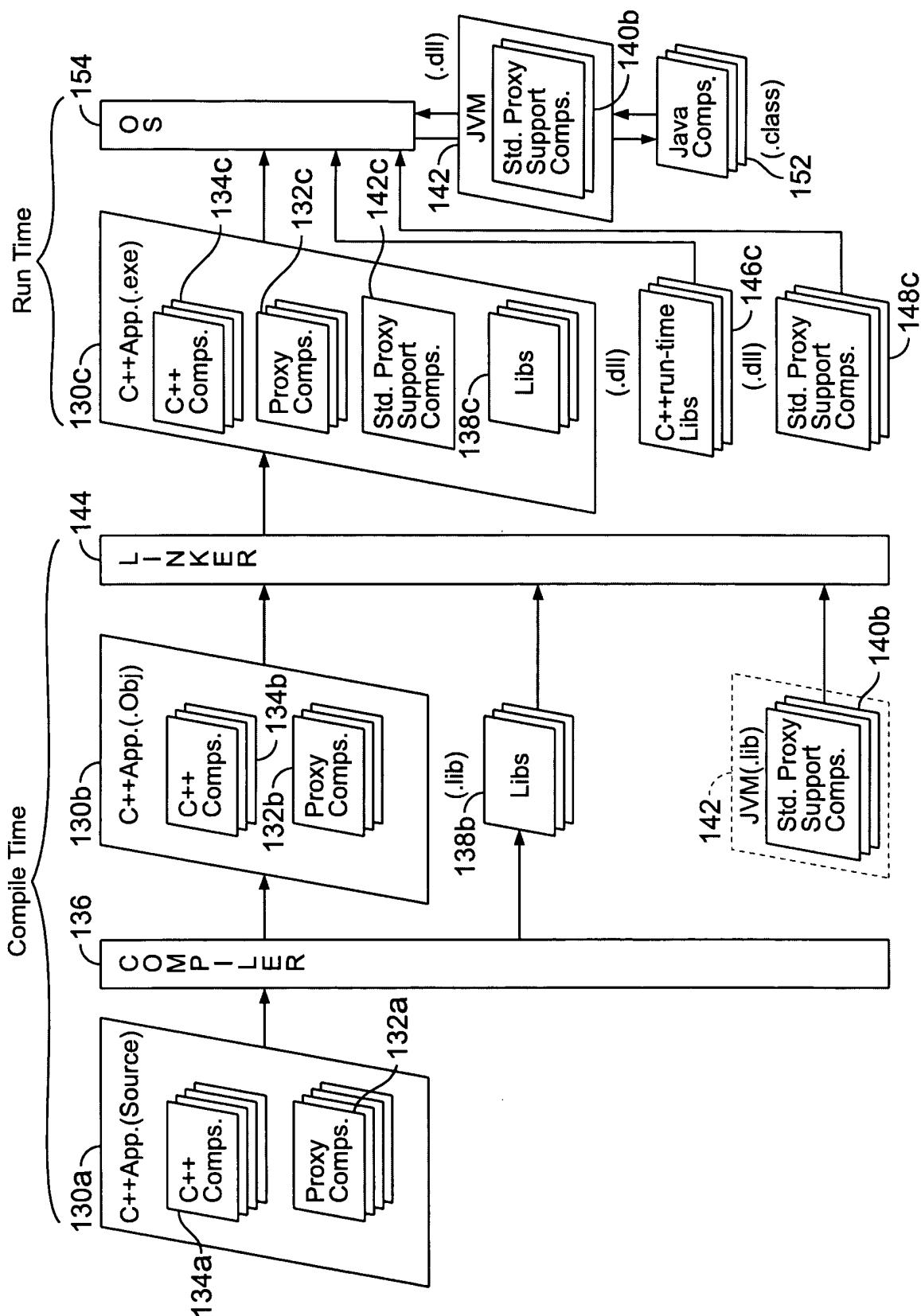


FIG. 18

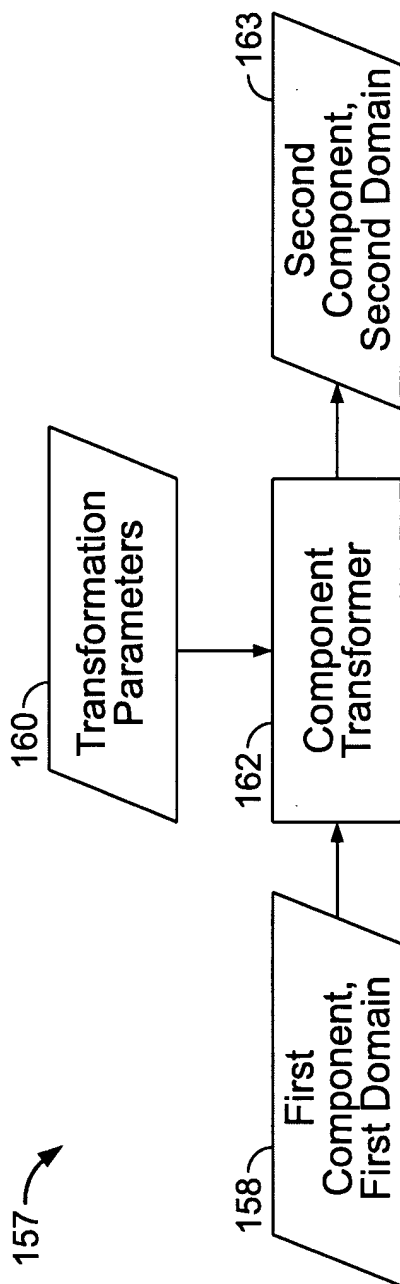


FIG. 19



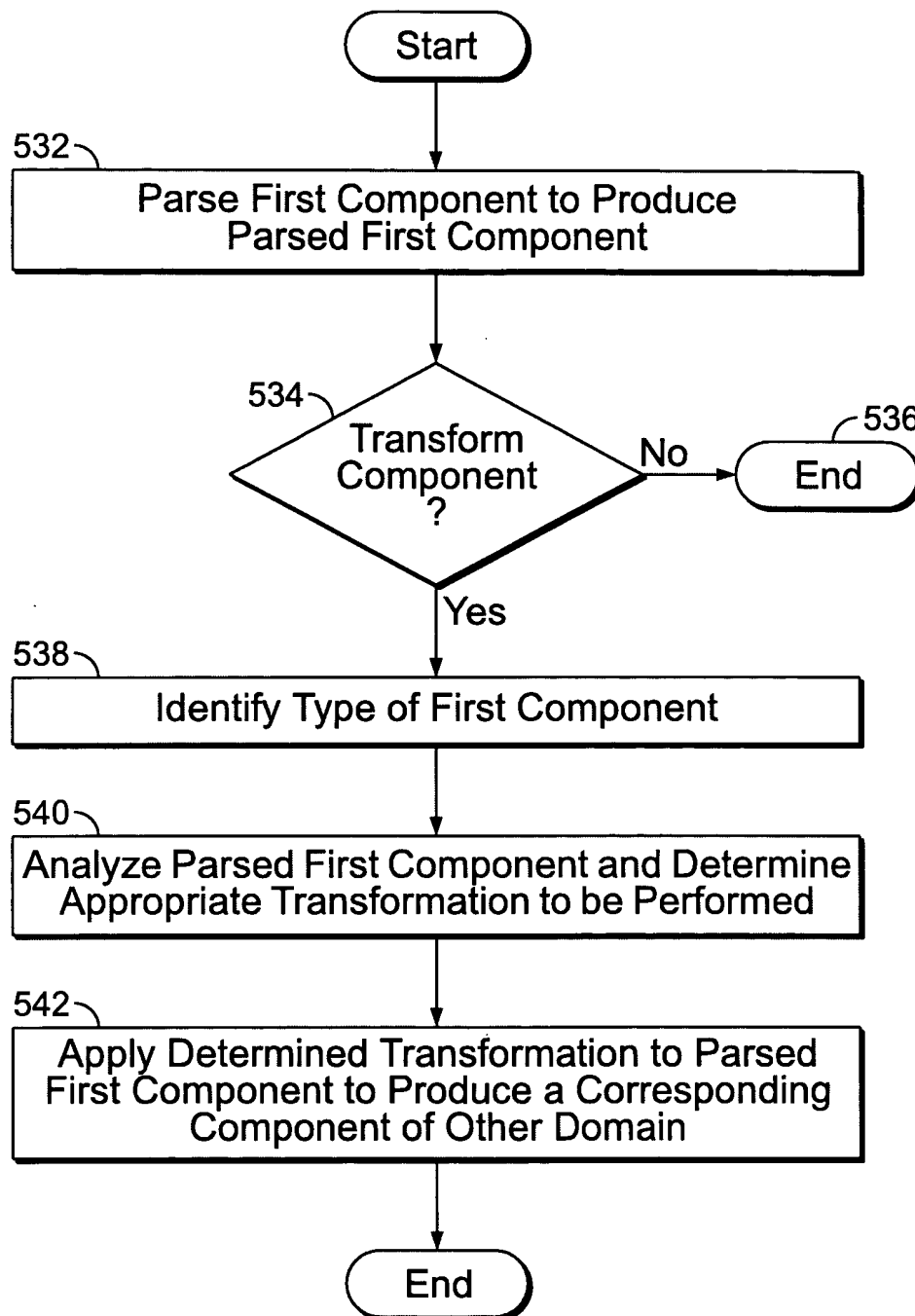
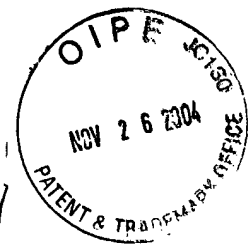
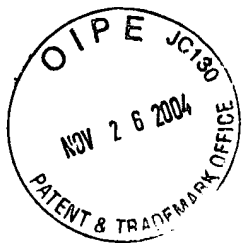


FIG. 20



540

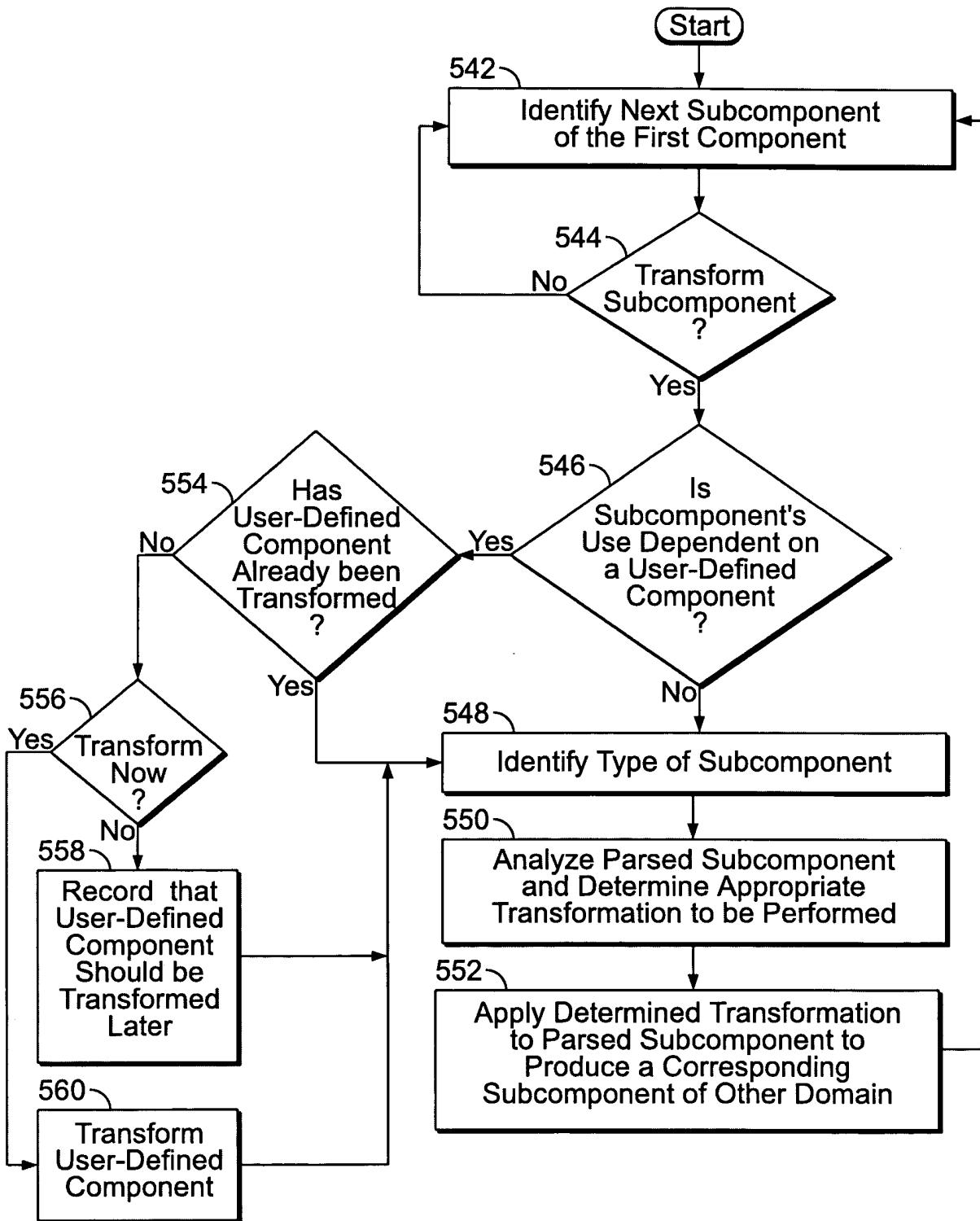


FIG. 21

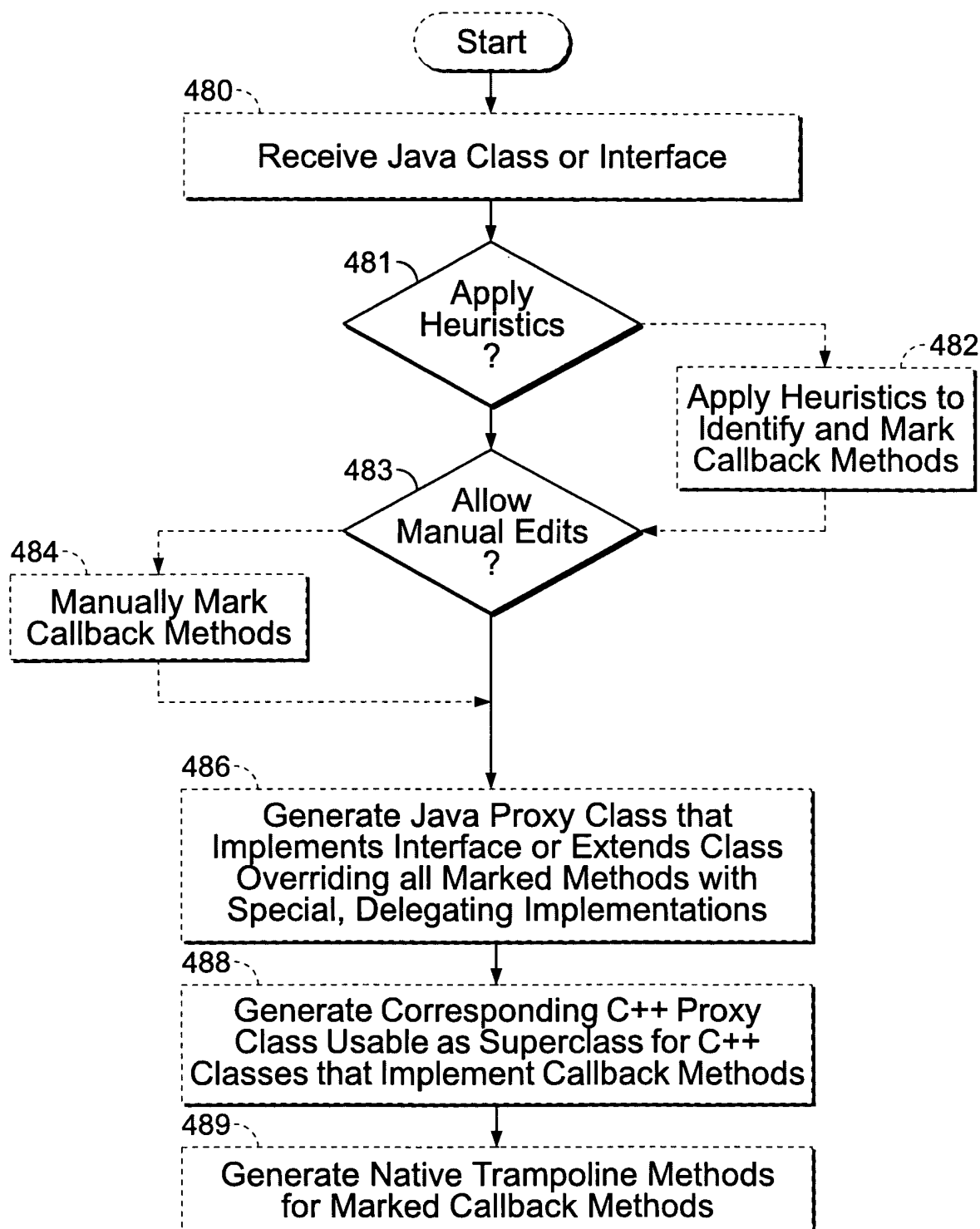


FIG. 22

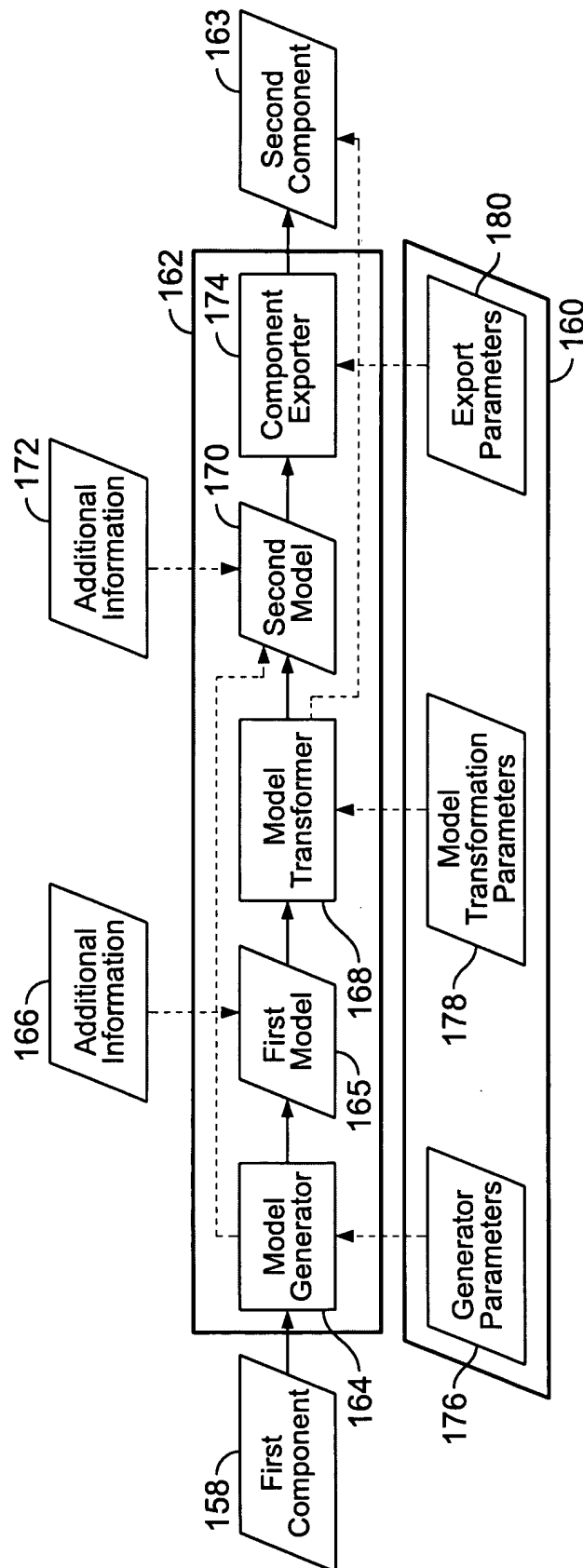


FIG. 23



300a

```
public class Counter implements java.io.Serializable
{
    public static final int UP = 1;
    public static final int DOWN = 2;

    private int max;
    private int direction;

    //creates a new UP-counter with the given maximum
    public Counter( int _max )
    {
        this( _max, UP );
    }

    //creates a new counter with given maximum and direction
    public Counter( int _max, int _direction )
    {
        max = _max;
        direction = _direction;
    }

    //counts in the direction specified and outputs the numbers
    public void count()
    {
        if( direction == UP )
            for( int l=0; l<max; l++ )
                System.out.println( " " + l );
        else if ( direction == DOWN )
            for ( int l=max-1; l>=0; l-- )
                System.out.println( " " + l );
    }

    //returns true if this instance is an UP counter
    public boolean isUpCounter()
    {
        return ( direction == UP );
    }

    //returns the maximum of the counter
    public final int getMax()
    {
        return max;
    }

    //creates a counter with the same maximum as this counter, but reverse direction
    public Counter getReverseCounter()
    {
        return new Counter( max, direction == UP ? DOWN : UP );
    }
}
```

FIG. 24

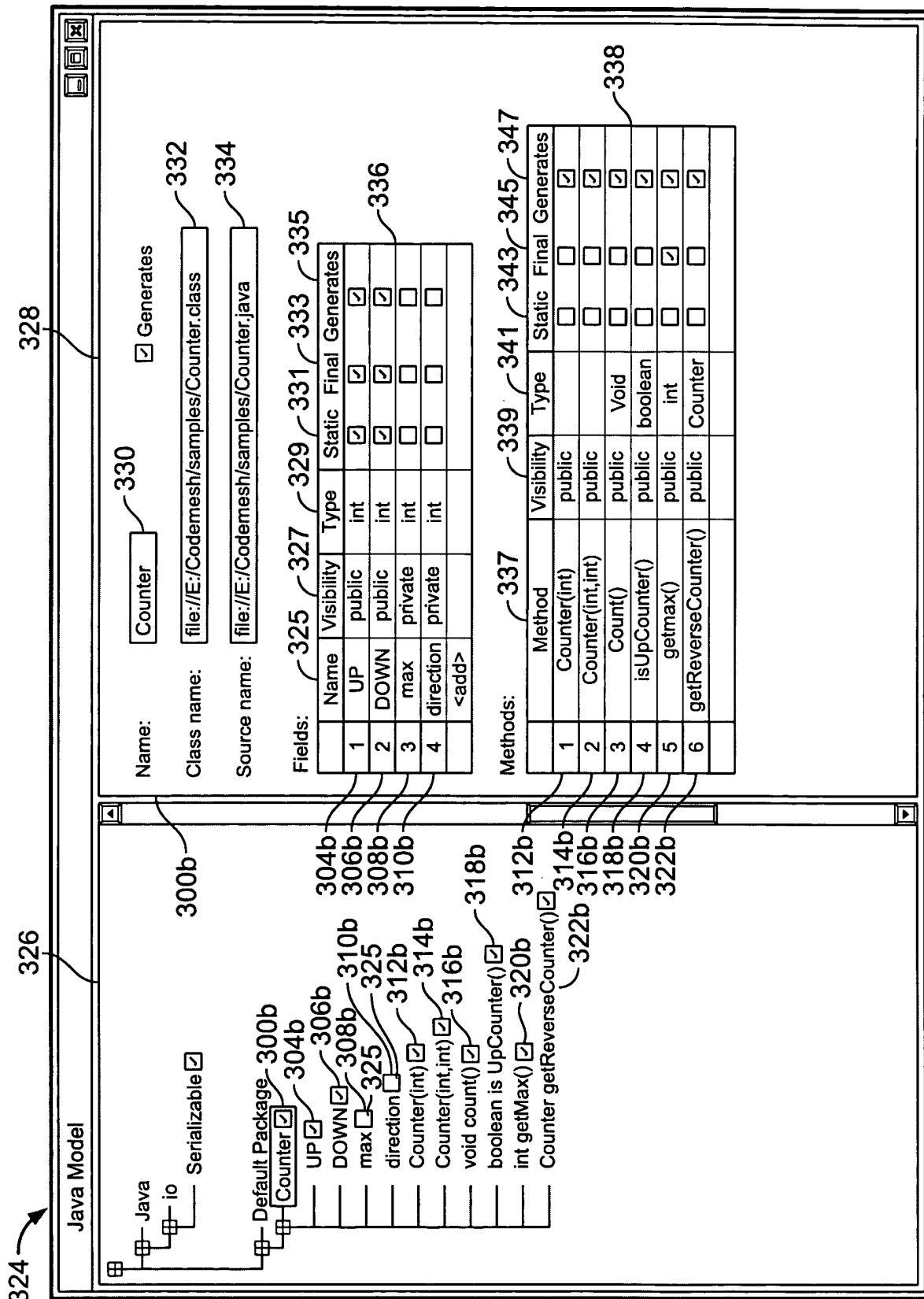
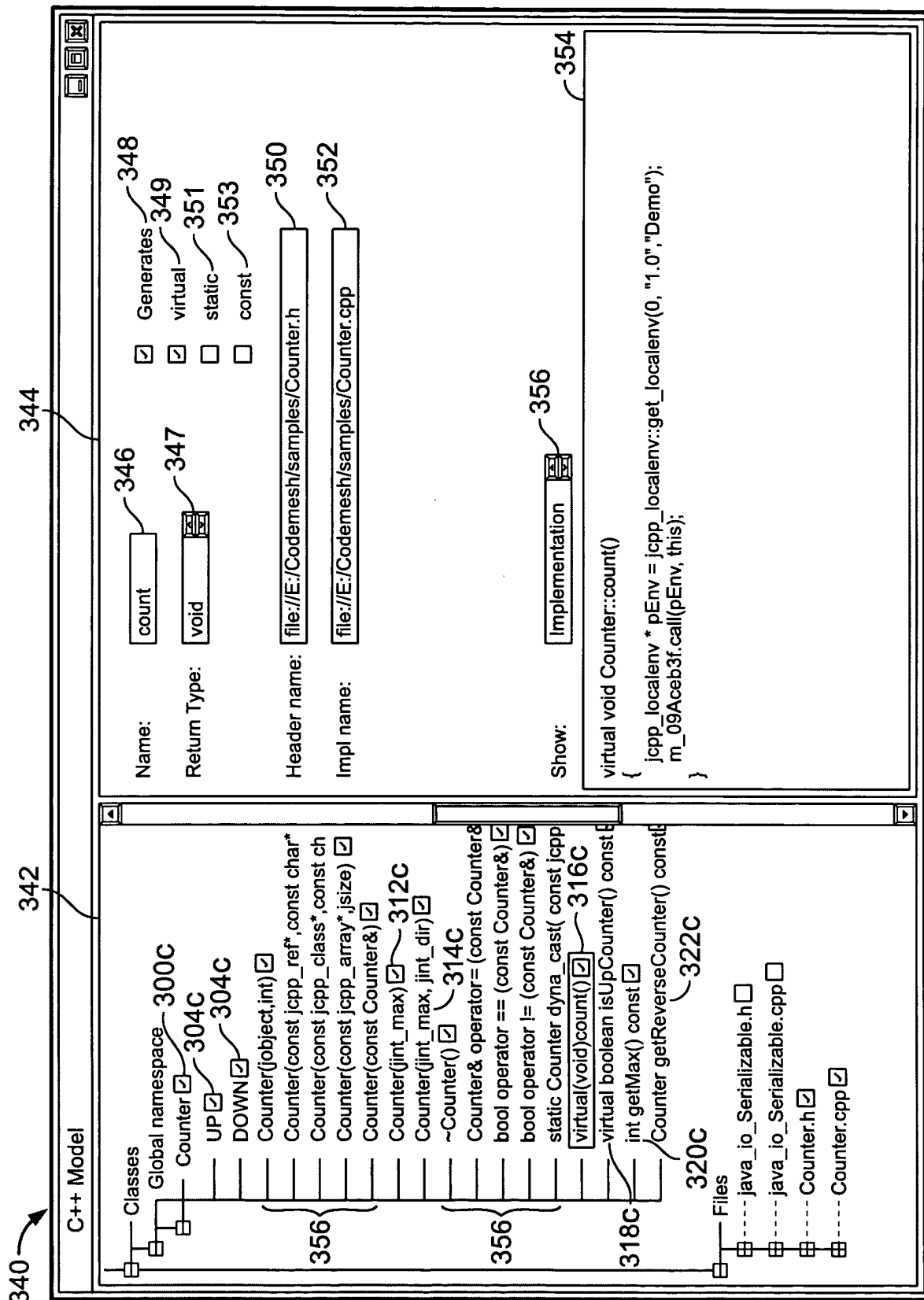


FIG. 25





600 →

## Bottom-Up Port By Proxy (Initial)

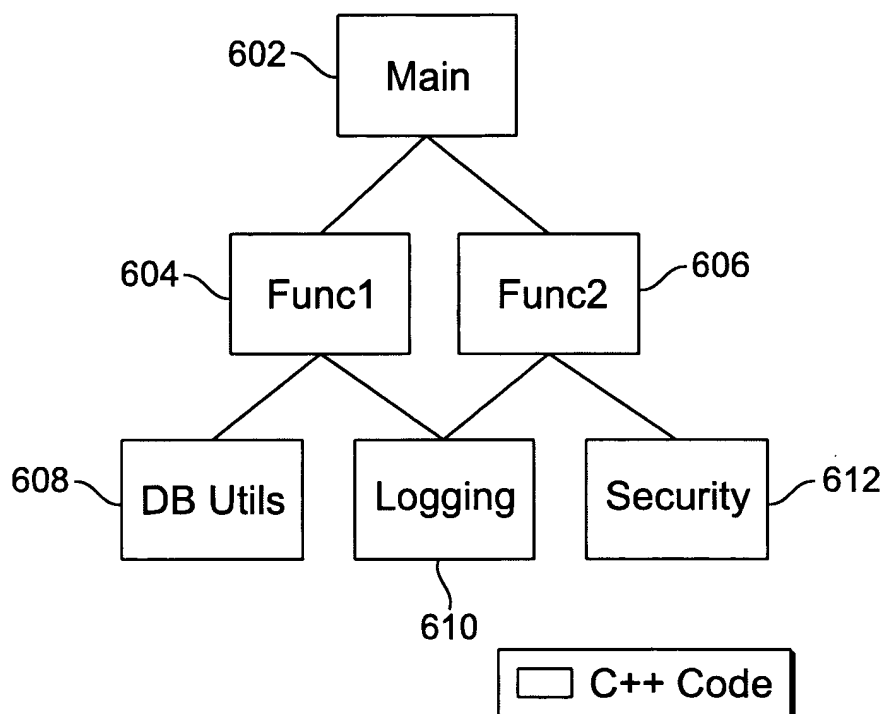
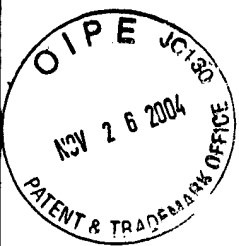


FIG. 27A





600 →

## Bottom-Up Port By Proxy (1st Step)

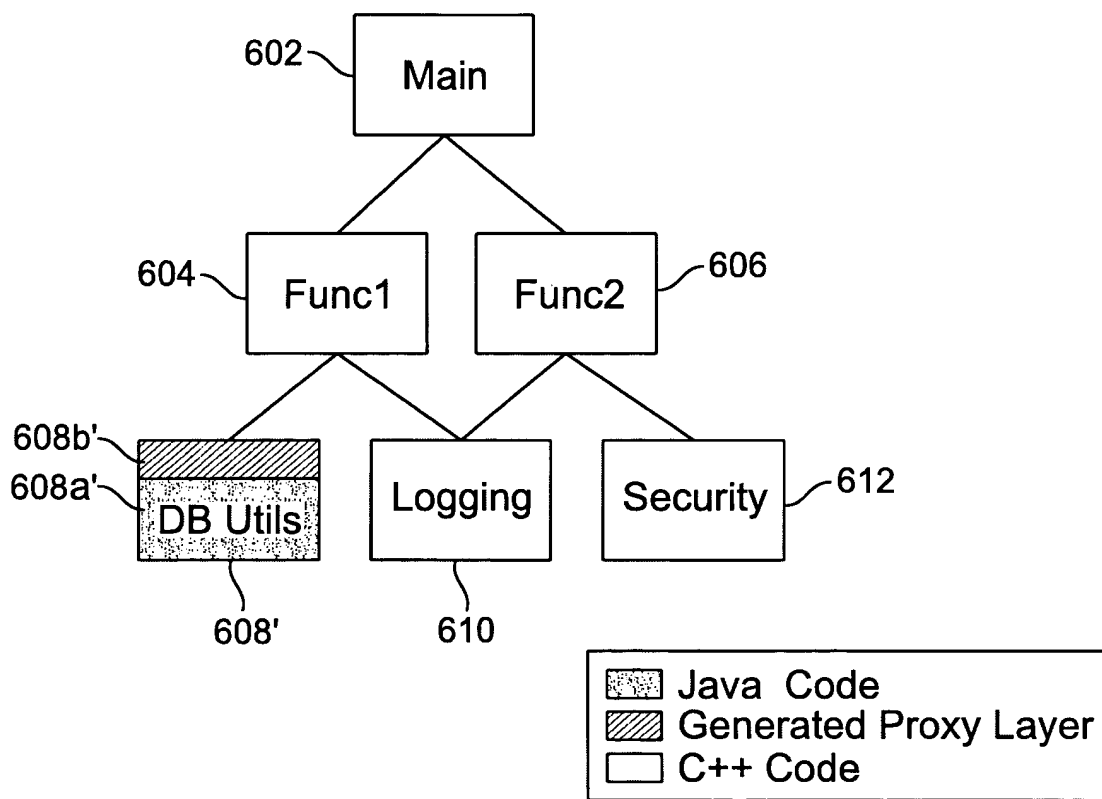


FIG. 27B



600

### Bottom-Up Port By Proxy (2nd Step)

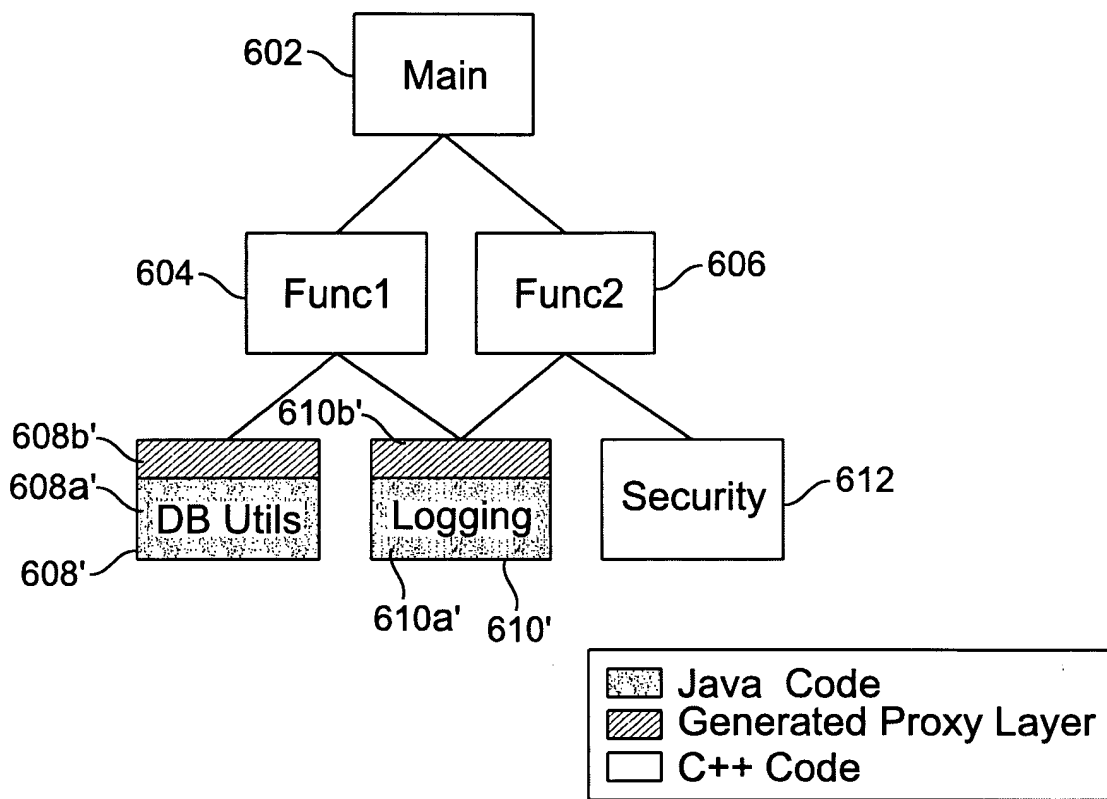


FIG. 27C



600

### Bottom-Up Port By Proxy (3rd Step)

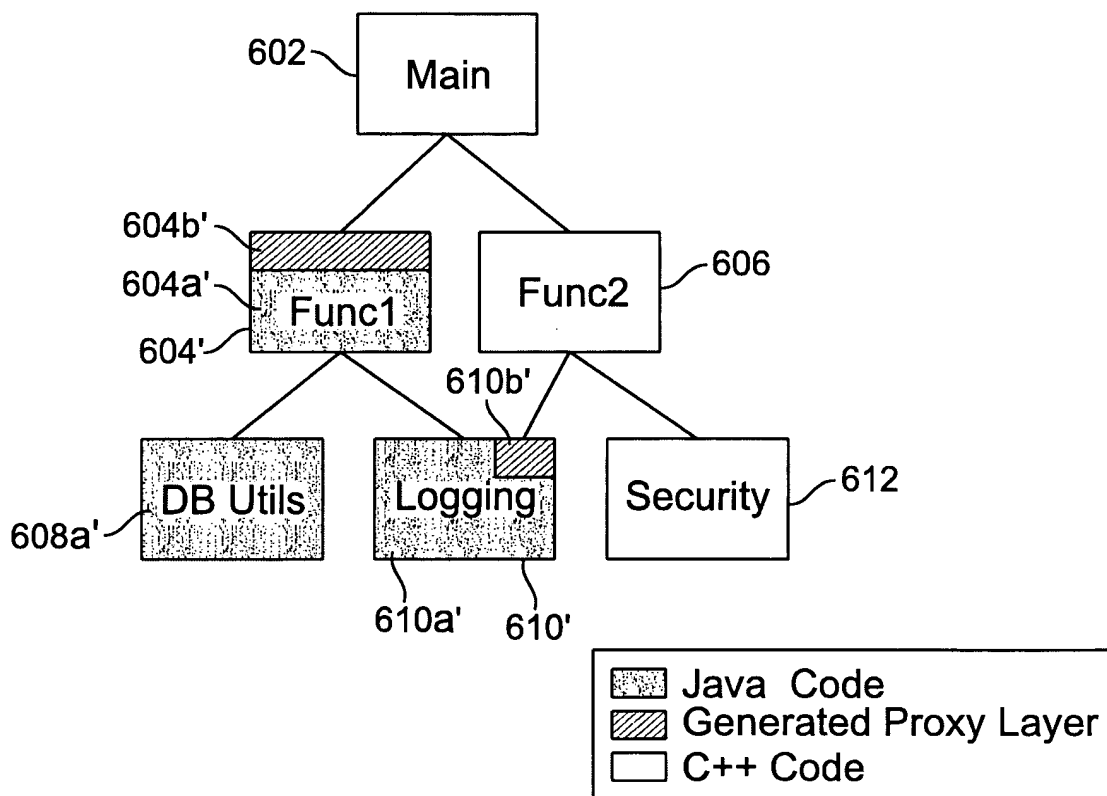
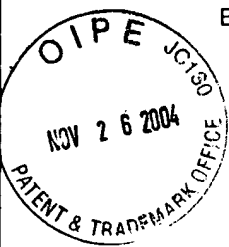


FIG. 27D



600

### Bottom-Up Port By Proxy (4th Step)

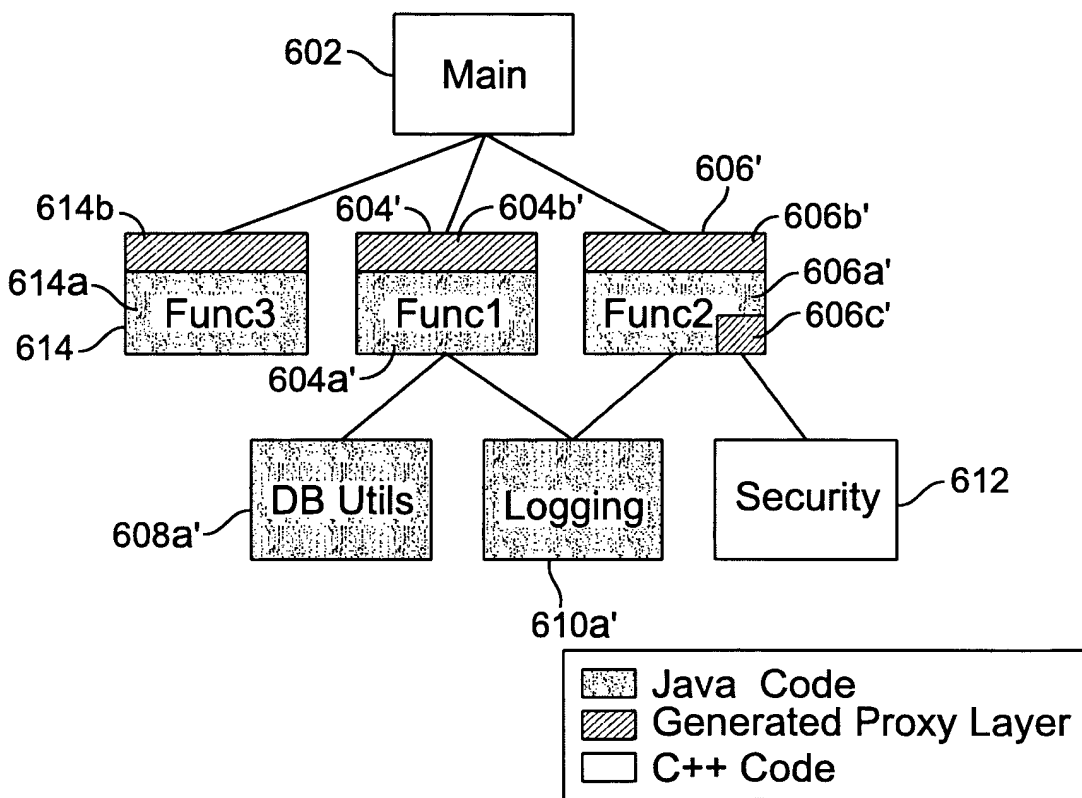
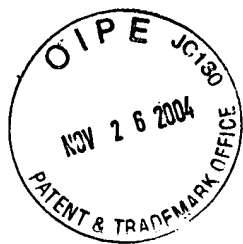


FIG. 27E



600

### Bottom-Up Port By Proxy (5th Step)

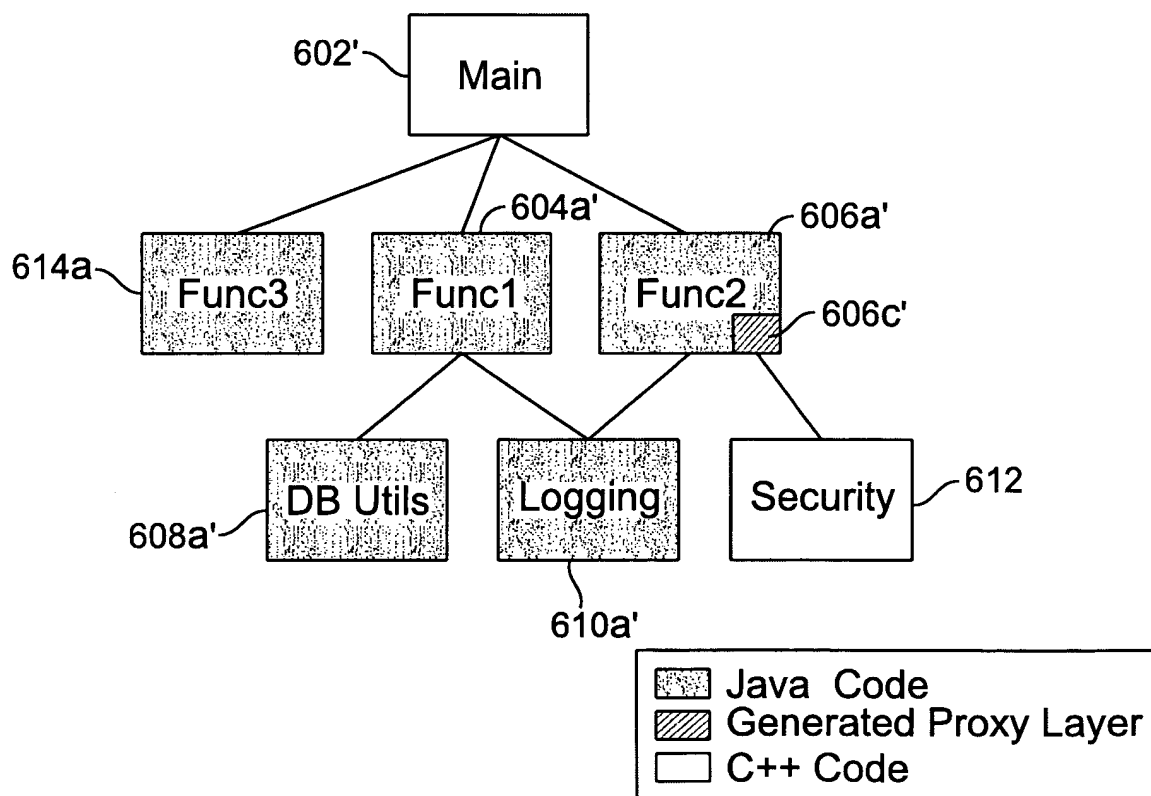


FIG. 27F